

# DynDNS bei Hetzner mit Dream Machine SE

## Was wollen wir?

Automatisiertes DynDNS bei Hetzner mit der Dream Machine SE.

## Warum wollen wir das?

Weil bei der Dream Machine SE beispielsweise Hetzner, aber auch viele andere Anbieter, kein vordefiniertes DynDNS Profil besitzen. Mit diesem Skript ist es über die API möglich weitere Anbieter nutzen zu können.

## Und wie geht das genau?

Ich habe eine Dream Machine SE und bin mit meiner Domain von Strato zu Hetzner gewechselt. Somit stand ich vor dem Problem, wie ich denn automatisiert DynDNS nutzen kann. Ich habe dazu ein Skript auf der Dream Machine SE angelegt und lasse dieses stündlich (außer zur geplanten IP-Neuzuweisung, da minütlich) laufen. Was dort genau passiert, wird im Quellcode deutlich. Benötigt wird der API-Token, die Zone-ID, sowie die A-Record-ID von Hetzner.

Beim Aufruf des Skriptes wird das WAN Interface lokal abgefragt und die externe IP-Adresse abgerufen und mit der vorherigen verglichen. Sollte es eine neue externe IP geben, wird der A-Record bei Hetzner aktualisiert.

Bash

```
#!/bin/bash
```

```
# API-Token und Zone ID
API_TOKEN="<Hetzer-API Token>"
ZONE_ID="<Domain-ID>"
RECORD_ID="<A-Record-ID>"
```

```
# Pfad zur Datei, in der die letzte bekannte WAN-IP-Adresse gespeichert wird
IP_FILE="/root/last_wan_ip"
LOG_FILE="/var/log/dyndns.log"
```

```
# Funktion zum Loggen mit Zeitstempel
log() {
  echo "$(date --iso-8601=seconds) - $1" >> "$LOG_FILE"
}
```

```
# Externe IP-Adresse abrufen
wan=$(/sbin/ifconfig ppp0 | /bin/grep inet | /usr/bin/awk -v OFS="\n" '{ print $2 }')
log "Aktuelle WAN-IP: $wan"
```

```
# Überprüfen, ob die Variable wan eine gültige IP-Adresse enthält
if [ -z "$wan" ]; then
  log "Fehler: Die Variable wan ist leer."
  exit 1
fi
```

```
if ! echo "$wan" | grep -Eq '^[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+$'; then
  log "Fehler: Die Variable wan enthält keine gültige IP-Adresse."
  exit 1
fi
```

```
# Letzte bekannte WAN-IP-Adresse lesen
if [ -f "$IP_FILE" ]; then
  last_wan=$(cat "$IP_FILE")
else
  last_wan=""
fi
log "Letzte bekannte WAN-IP: $last_wan"
```

```
# Prüfen, ob sich die IP-Adresse geändert hat
if [ "$wan" != "$last_wan" ]; then
  log "WAN-IP hat sich geändert von $last_wan auf $wan"
```

```
# DNS-Record aktualisieren
RESPONSE=$(curl -s -X "PUT" "https://dns.hetzner.com/api/v1/records/$RECORD_ID" \
-H "Content-Type: application/json" \
-H "Auth-API-Token: $API_TOKEN" \
-d "{
  \"value\": \"$wan\",
  \"ttl\": 86400,
  \"type\": \"A\",
  \"name\": \"@\",
  \"zone_id\": \"$ZONE_ID\"
}")
```

```
# Ergebnis prüfen
if echo "$RESPONSE" | grep -q '"error"'; then
  log "Fehler beim Aktualisieren des DNS-Records: $RESPONSE"
  exit 1
else
  log "DNS-Record erfolgreich aktualisiert."
```

Alles anzeigen

Das Skript wird im /root/ Verzeichnis angelegt und muss auf jeden Fall ausführbar gemacht werden. (chmod +x /root/<skript.sh>)

Anschließend einen Eintrag im Verzeichnis /etc/cron.d/<croneintrag> erstellen. Der Inhalt lautet wird folgt:

Code

```
50-59 3 * * * root sh /root/dyndns.sh >> /var/log/messages
1-10 4 * * * root sh /root/dyndns.sh >> /var/log/messages
0 */1 * * * root sh /root/dyndns.sh >> /var/log/messages
```

Außerdem werden sämtliche Vorgänge in /var/log/dyndns.log protokolliert.

Auswählen: \_\_\_\_\_

Gültige Software-Version Keine Firmware-Relevanz!